

A ROBUST SVD-FREE APPROACH TO MATRIX COMPLETION, WITH APPLICATIONS TO INTERPOLATION OF LARGE SCALE DATA

ALEKSANDR ARAVKIN ^{*}, RAJIV KUMAR [†], HASSAN MANSOUR [‡], BEN RECHT [§], AND FELIX J. HERRMANN [¶]

Abstract. Recent SVD-free matrix factorization formulations have enabled rank minimization for systems with millions of rows and columns, paving the way for matrix completion in extremely large-scale applications, such as seismic data interpolation.

In this paper, we consider matrix completion formulations designed to hit a target data-fitting error level provided by the user, and propose an algorithm that is able to exploit factorized formulations to solve the corresponding optimization problem. Since practitioners typically have strong prior knowledge about target error level, this innovation makes it easy to apply the algorithm in practice, leaving only the factor rank to be determined. We explore the role that rank of the factors plays in our formulation, and show that rank can be easily adjusted as the inversion proceeds.

Within the established framework, we then propose two extensions that are highly relevant to solving practical challenges of data interpolation. First, we propose a weighted extension that allows known subspace information to improve the results of matrix completion formulations. We show how this weighting can be used in the context of frequency continuation, an essential aspect to seismic data interpolation. Second, we extend matrix completion formulations to be extremely robust to large measurement errors in the available data.

We illustrate the advantages of the basic approach on the Netflix Prize problem using the Movielens (1M) dataset. Then, we use the new method, along with its robust and subspace re-weighted extensions, to obtain high-quality reconstructions for large scale seismic interpolation problems with real data, even in the presence of extreme data contamination.

1. Introduction. Sparsity- and rank-regularization have had tremendous impact in many areas over the last several decades. Sparsity in certain transform domains has been exploited to solve underdetermined linear systems with applications to compressed sensing [9, 8], natural image denoising/inpainting [34, 22, 24], and seismic image processing [17, 27, 16, 25]. Analogously, low rank structure has been used to efficiently solve matrix completion problems, including the Netflix problem, along with many other applications, including control, system identification, signal processing, and combinatorial optimization [10, 29, 7], and seismic data interpolation and denoising [28].

Regularization formulations for both types of problems introduce a regularization functional of the decision variable, either by adding an explicit penalty to the data-fitting term

$$\min_x \rho(\mathcal{A}(x) - b) + \lambda \|x\|, \quad (\text{QP}_\lambda)$$

or by imposing constraints

$$\min_x \rho(\mathcal{A}(x) - b) \quad \text{s.t.} \quad \|x\| \leq \tau. \quad (\text{LASSO}_\tau)$$

In these formulations, x may be either a matrix or a vector, $\|\cdot\|$ may be a sparsity or rank promoting penalty such as $\|\cdot\|_1$ or $\|\cdot\|_*$, \mathcal{A} may be any linear operator that predicts the observed data vector b of size $p \times 1$, and $\rho(\cdot)$ is typically taken to be the 2-norm.

These approaches require the user to provide regularization parameters whose values are typically not known ahead of time, and otherwise require fitting or cross-validation procedures.

The alternate formulation

$$\min_x \|x\| \quad \text{s.t.} \quad \rho(\mathcal{A}(x) - b) \leq \eta. \quad (\text{BPDN}_\eta)$$

has been successfully used for the sparse regularization of large scale systems [4], and proposed for nuclear norm regularization [5]. This formulation requires the user to provide an acceptable error in the data fitting domain (BPDN_η), and is preferable for many applications, especially when practitioners know (or are able to estimate) an approximate data error level.

^{*}IBM T.J. Watson Research Center, Yorktown Heights, NY 10598, USA (saravkin@us.ibm.com).

[†]Department of Earth and Ocean Sciences, UBC, Vancouver, BC, Canada (rakumar@eos.ubc.ca)

[‡]Department of Mathematics, UBC, Vancouver, BC, Canada (mansour.hassan@gmail.com)

[§]Department of Computer Science, University of Wisconsin-Madison, Madison, WI, USA (brecht@cs.wisc.edu)

[¶]Department of Earth and Ocean Sciences, UBC, Vancouver, BC, Canada (fherrmann@eos.ubc.ca)

A practical implementation of (BPDN_η) for large scale matrix completion problems is difficult because of the tremendous size of the systems of interest, which makes SVD-based approaches intractable. Fortunately, a growing literature on factorization-based rank optimization approaches has enabled matrix completion formulations for (QP_λ) and (LASSO_τ) approaches for extremely large-scale systems that avoiding costly SVD computations [31, 21, 30]. In this paper, we extend the framework of [5] to incorporate matrix factorization ideas, enabling the (BPDN_η) formulation for rank regularization of extremely large scale problems, such as seismic data interpolation.

In factorized formulations, the user must specify a factor rank. From a computational perspective, it is better that the rank stay small; however if it is too small, it may be impossible to solve (BPDN_η) to a specified error level η . Fortunately, factor rank can be adjusted on the fly within the framework we develop.

While formulations in [4, 5] choose ρ in (BPDN_η) to be the quadratic penalty, recent extensions [1] allow more general penalties to be used. In particular, robust convex (see e.g. [18]) and nonconvex penalties (see e.g. [19, 2]) can be used to measure misfit error in the (BPDN_η) formulation. These extensions are also captured by the framework we propose for the rank optimization problem, and allow matrix completion problems that are robust to data contamination.

Finally, subspace information can be used to inform the matrix completion problem, analogously to how partial support information can be used to improve the sparse recovery problem [12]. This idea is especially important for seismic interpolation, where frequency continuation is used. We show that subspace information can be incorporated into the proposed framework using reweighting.

To summarize, we design factorization-based formulations and algorithms for matrix completion that

1. Achieve a specified target misfit level provided by the user (i.e. solve (BPDN_η)).
2. Achieve recovery in spite of severe data contamination using robust cost functions ρ in (BPDN_η)
3. Incorporate subspace information into the inversion using re-weighting.

The paper proceeds as follows. In section 2, we briefly discuss and compare the formulations (QP_λ) , (LASSO_τ) , and (BPDN_η) . We also review the SPGL_1 algorithm [4] to solve (BPDN_η) , along with recent extensions for (BPDN_η) formulations developed in [1]. In section 3, we formulate the convex relaxation for the rank optimization problem, and review SVD-free factorization methods. In section 4, we extend analysis from [6] to characterize the relationship between local minima of rank-optimization problems and their factorized counterparts in a very general setting. In section 5, we propose an algorithm that combines matrix factorization with the approach proposed by [4] and extended in [1]. We develop the robust extensions in section 6, and reweighting extensions in section 7. Numerical results for both the Netflix Prize problem and for seismic trace interpolation of real data are presented in section 8.

2. Regularization formulations. Each of the three formulations (QP_λ) , (LASSO_τ) , and (BPDN_η) controls the tradeoff between data fitting and a regularization functional using a regularization parameter. However, there are important differences between them.

From an optimization perspective, most algorithms solve (QP_λ) or (LASSO_τ) , together with a continuation strategy to modify τ or λ , see e.g., [11, 4]. However, from a modeling perspective (BPDN_η) has a significant advantage, since the η parameter can be directly interpreted as a noise ceiling. In many applications, such as seismic data interpolation, scientists have good prior knowledge of the noise floor, or a threshold beyond which noise is commensurate with the data. In the absence of such knowledge, one still wants an algorithm that returns a reasonable solution given a fixed computational budget.

van den Berg and Friedlander [4] proposed the SPGL_1 algorithm for optimizing (BPDN_η) that captures the features discussed above. Their approach allows solving (BPDN_η) using a series of inexact solutions to (LASSO_τ) . The bridge between these problems is provided by the *value function*

$$v(\tau) = \min_x \rho(\mathcal{A}(x) - b) \quad \text{s.t. } \|x\| \leq \tau, \quad (2.1)$$

where the particular choice of $\rho(\cdot) = \|\cdot\|^2$ was made in [4, 5]. The (BPDN_η) problem can be solved by find the root of $v(\tau) = \eta$ using Newton's method:

$$\tau^{k+1} = \tau^k - \frac{v(\tau) - \eta}{v'(\tau)}, \quad (2.2)$$

and the quantities $v(\tau)$ and $v'(\tau)$ can be approximated by solving (LASSO_τ) problems. The graph of $v(\tau)$ is often called the Pareto curve. In the context of sparsity optimization, (BPDN_η) and (LASSO_τ) are known to be equivalent for certain values of parameters τ and η . Recently, these results were extended to a very broad class of formulations (see [1, Theorem 2.1])¹.

For any ρ , $v(\tau)$ is non-increasing, since larger τ allow a bigger feasible set. For any convex ρ in (2.1), $v(\tau)$ is convex by inf-projection [32, Proposition 2.22]. When ρ is also differentiable, it follows from [1, Theorem 5.2] that $v(\tau)$ is differentiable, with derivative given in closed form by

$$v'(\tau) = -\|\mathcal{A}^*\nabla\rho(b - \mathcal{A}\bar{x})\|_d, \quad (2.3)$$

where \mathcal{A}^* is the adjoint to the operator \mathcal{A} , $\|\cdot\|_d$ is the dual norm to $\|\cdot\|$, and \bar{x} solves LASSO_τ . For example, when the norm $\|\cdot\|$ in (2.1) is the 1-norm, the dual norm is the infinity norm, and (2.3) evaluates to the maximum absolute entry of the gradient. In the matrix case, $\|\cdot\|$ is typically taken to be the nuclear norm, and then $\|\cdot\|_d$ is the spectral norm, so (2.3) evaluates to the maximum singular value of $\mathcal{A}^*\nabla\rho(r)$.

To design effective optimization methods, one has to be able to evaluate $v(\tau)$, and to compute the dual norm $\|\cdot\|_d$. Evaluating $v(\tau)$ requires solving a sequence of optimization problems (2.1), for the sequence of τ given by (2.2). These problems can be solved inexactly, with increasing accuracy as the optimization proceeds [4]. For large scale systems, the method of choice is typically a first-order method, such as spectral projected gradient. Fast projection is therefore a necessary requirement for tractable implementation, since it is used in every iteration of every subproblem.

With the inexact strategy, the convergence rate of the Newton iteration (2.2) may depend on the conditioning of the linear operator \mathcal{A} [4, Theorem 3.1]. For well-conditioned problems, in practice one can often observe only a few (6-10) (LASSO_τ) problems to find the solution for (BPDN_η) for a given η . As the optimization proceeds, (LASSO_τ) problems for larger τ warm-start from the solution corresponding to the previous τ .

3. Factorization approach to rank optimization. We now consider (BPDN_η) in the specific context of rank minimization. In this setting, $\|\cdot\|$ taken to be the nuclear norm, where for a matrix $X \in \mathbb{R}^{n \times m}$, $\|X\|_* = \|\sigma\|_1$, where σ is the vector of singular values. The dual norm in this case is σ_∞ , which is relatively easy to find for very large systems.

Much more difficult is solving the optimization problem in (2.1). For the large system case, this would require projecting onto the set $\|X\|_* \leq \tau$, which requires repeated application of SVDs. This is not feasible for large systems.

Factorization-based approaches allow matrix completion for extremely large-scale systems by avoiding costly SVD computations [31, 20, 30]. The main idea is to parametrize the matrix X as a product,

$$X = LR^T, \quad (3.1)$$

and to optimize over the factors L, R . If $X \in \mathbb{R}^{n \times m}$, then $L \in \mathbb{R}^{n \times k}$, and $R \in \mathbb{R}^{m \times k}$. The decision variable therefore has dimension $k(n+m)$, rather than nm ; giving tremendous savings when $k \ll m, n$.

Aside from the considerable savings in the size of the decision variable, the factorization technique gives an additional advantage: it allows the use of a factorization identities to make the projection problem in (LASSO_τ) trivial, entirely avoiding the SVD.

For the nuclear norm, we have [31]

$$\|X\|_* = \inf_{X=LR^T} \frac{1}{2} \left\| \begin{bmatrix} L \\ R \end{bmatrix} \right\|_F^2. \quad (3.2)$$

Working with a particular representation $X = LR^T$, therefore, guarantees that

$$\|X\|_* = \|LR^T\|_* \leq \frac{1}{2} \left\| \begin{bmatrix} L \\ R \end{bmatrix} \right\|_F^2. \quad (3.3)$$

The nuclear norm is not the only formulation that can be factorized. [21] have recently introduced the max norm, which is closely related to the nuclear norm and has been successfully used for matrix completion.

¹Surprisingly, convexity of misfit and regularizer is not required for the result.

4. General Burer-Monteiro Result. All of the algorithms we propose for matrix completion are based on the factorization approach described above. Even though the change of variables $X = LR^T$ makes the problem nonconvex, it turns out that for a surprisingly general class of problems, the change of variables does not introduce any extraneous local minima, and in particular any local minimum of the factorized problem corresponds to a local or global minimum of a related un-factorized problem. This result appeared in [6, Proposition 2.3] in the context of SDP; however, it holds in full generality, as the authors point out [6, p. 431].

Here, we state the result in full generality for a broad class of problems, which is general enough to capture all of our formulations of interest. For completeness, we provide a proof in the appendix.

THEOREM 4.1 (General Factorization Theorem). *Consider an optimization problem of the form*

$$\begin{aligned} \min_{Z \succeq 0} \quad & f(Z) \\ \text{s.t.} \quad & g_i(Z) \leq 0 \quad i = 1, \dots, n \\ & h_j(Z) = 0 \quad j = 1, \dots, m \\ & \text{rank}(Z) \leq r, \end{aligned} \tag{4.1}$$

where $Z \in \mathbb{R}^{n \times n}$ is positive semidefinite, and f, g_i, h_i are continuous. Using the change of variable $Z = SS^T$, take $S \in \mathbb{R}^{n \times r}$, and consider the problem

$$\begin{aligned} \min_S \quad & f(SS^T) \\ \text{s.t.} \quad & g_i(SS^T) \leq 0 \quad i = 1, \dots, n \\ & h_j(SS^T) = 0 \quad j = 1, \dots, m \end{aligned} \tag{4.2}$$

Consider $\bar{Z} = \bar{S}\bar{S}^T$, where \bar{Z} is feasible for (4.1). Then \bar{Z} is a local minimum of (4.1) if and only if \bar{S} is a local minimum of (4.2).

In order to apply Theorem 4.1 to the formulations of interest, we need to show that these formulations can be expressed in terms of a positive semidefinite matrix Z . The connection comes from the SDP characterization of the nuclear norm.

It was shown in [29, Sec. 2] that the nuclear norm admits a semi-definite programming (SDP) formulation. Given a matrix $X \in \mathbb{R}^{n \times m}$, we can write $\|X\|_*$ using an auxiliary matrix $Z \in \mathbb{R}^{(n+m) \times (n+m)}$

$$\begin{aligned} \|X\|_* &= \min_{Z \succeq 0} \frac{1}{2} \text{Tr}(Z) \\ \text{subject to } Z_{1,2} &= Z_{2,1}^T = X, \end{aligned} \tag{4.3}$$

where $Z_{1,2}$ is the upper right $n \times m$ block of Z , and $Z_{2,1}$ is the lower left $m \times n$ block. More precisely, the matrix Z is a symmetric positive semidefinite matrix having the structure

$$Z = \begin{bmatrix} LL^T & X \\ X^T & RR^T \end{bmatrix}, \tag{4.4}$$

where L and R have the same rank as X , and $\text{Tr}(Z) = \|L\|_F^2 + \|R\|_F^2$.

Using this characterization, we can show that a broad class of formulations of interest in this paper are in fact problems in the class characterized by Theorem 4.1.

COROLLARY 4.2 (General Matrix Lasso). *Any optimization problem of the form*

$$\begin{aligned} \min_X \quad & f(X) \\ \text{s.t.} \quad & \|X\|_* \leq \tau \end{aligned} \tag{4.5}$$

has an equivalent problem in the class of problems (4.1) characterized by Theorem 4.1.

Proof. Using (4.3), write (4.5) as

$$\begin{aligned} \min_{Z \succeq 0} \quad & f(\mathcal{R}(Z)) \\ \text{s.t.} \quad & \text{Tr}(Z) \leq \tau, \end{aligned} \tag{4.6}$$

where $\mathcal{R}(Z)$ extracts the upper right $n \times m$ block of Z .

□

5. Factorized Pareto Algorithm. The factorized formulations in the previous section have been used to design several algorithms for large scale matrix completion and rank minimization [21, 30]. However, all of these formulations take the form (QP_λ) or (LASSO_τ) that have the disadvantage of having to identify the parameters λ and τ ahead of the optimization procedure.

Instead, we propose to use the factorized formulations to solve the (BPDN_η) problem by traversing the Pareto curve of the nuclear norm minimization problem. In particular, we integrate the factorization procedure into the SPG_{ℓ_1} framework, which allows to find the minimum rank solution by solving a sequence of factorized (LASSO_τ) subproblems (5.2). The cost of solving the factorized (LASSO_τ) subproblems is relatively cheap and the resulting algorithm takes advantage of the inexact subproblem strategy in [4].

For the classic nuclear norm minimization problem, we define

$$v(\tau) = \min_X \|\mathcal{A}(X) - b\|_2^2 \quad \text{s.t.} \quad \|X\|_* \leq \tau, \quad (5.1)$$

and find $v(\tau) = \eta$ using the iteration (2.2).

However, rather than parameterizing our problem with X , which requires SVD for the projection, we use the factorization formulation, exploiting Theorem 4.1 and Corollary 4.2. When evaluating the value function $v(\tau)$, we actually solve

$$\min_{L,R} \|\mathcal{A}(LR^T) - b\|_2^2 \quad \text{s.t.} \quad \frac{1}{2} \left\| \begin{bmatrix} L \\ R \end{bmatrix} \right\|_F^2 \leq \tau. \quad (5.2)$$

By Theorem 4.1 and Corollary 4.2, any local solution to this problem corresponds to a local solution of the true LASSO problem, subject to a rank constraint on X . For any convex ρ , every local minimum of (LASSO_τ) is also a global minimum, though it is not necessarily unique. When the rank of the factors L and R is smaller than the rank of the optimal LASSO solution, the algorithm looks for local minima of the rank-constrained LASSO problem.

The problem (5.2) is optimized using the spectral projected gradient algorithm. The gradient is easy to compute, and the projection requires rescaling all entries of L, R by a single value, which is fast, simple, and parallelizable.

To evaluate $v'(\tau)$, we use the formula (2.3), treating $X = LR^T$ as a matrix in its own right. This requires computing the largest singular value of

$$\mathcal{A}^*(b - \mathcal{A}(\bar{L}\bar{R}^T)),$$

which can be done relatively quickly.

5.1. Increasing k on the fly. An important aspect to the factorization formulation is the choice of rank for the factors L, R in (3.1). The choice of rank k directly controls the rank of $X = LR^T$, and picking k too small makes it impossible to fit the data up to a target level η . Moreover, a large enough rank guarantees that the factorized formulation will find a global minimum to the LASSO problem. It is therefore desirable to be able to dynamically increase k , especially for problems with multiscale structure (e.g. seismic data, where k is expected to grow with frequency).

Fortunately, adding columns to L and R can be done on the fly, since

$$\begin{bmatrix} L & l \end{bmatrix} \begin{bmatrix} R & r \end{bmatrix}^T = LR^T + lr^T.$$

Moreover, the proposed framework for solving (BPDN_η) is fully compatible with this strategy, since the underlying root finding is blind to the factorization representation. Changing k only affects iteration (2.2) through $v(\tau)$ and $v'(\tau)$.

5.2. Computational efficiency. One way of assessing the cost of the factorized Pareto algorithm is to compare the computational cost per iteration of the factorization constrained LASSO subproblems (5.2) with that of the nuclear norm constrained LASSO subproblems (5.1). We first consider the cost for computing the gradient direction. While both methods must compute the action of \mathcal{A}^* on a vector, the factorized formulation must modify factors L, R (at a cost of $O(k(n+m))$) and re-form the matrix $X = LR^T$ (at a cost of at most $O(knm)$), for every iteration and line search evaluation. Note that when \mathcal{A} is a sampling matrix, it is sufficient to form only the entries of X that are sampled by \mathcal{A} , thus reducing the cost to $O(kp)$, where p is the dimension of the measurement vector b . The sparser the sampling operator \mathcal{A} , the greater the savings; however, the factorized approach incurs a cost proportional to the rank k of the factors. The standard approach must modify the iterate X , at a cost of $O(nm)$, for every iteration and line search evaluation. If the fraction sampled is smaller than the chosen rank k , the factorized approach is actually cheaper than the standard method.

We now consider the difference in cost involved in the projection. The main benefit for the factorized formulation is that projection is done using the Frobenius norm formulation (5.2), and so the cost is $O(k(n+m))$ for every projection. In contrast, any classic implementation that computes the full SVD in order to accomplish the projection (see e.g. [3]) is dominated by the cost of this calculation, which is $O(nm^2)$, assuming without loss of generality that $n \geq m$. This cost, incurred at every projection, becomes prohibitively expensive for large scale computation. Finally, both standard and factorized versions of the algorithm require computing the maximum singular value in order to compute $v'(\tau)$.

It can be seen that for a large number of projected gradient iterations, the cost of the nuclear norm formulation becomes significantly larger than our factorized formulation. Moreover, based on the analysis in section 4, if the rank of the factors L and R is at least equal to k , then the factorized LASSO subproblems are guaranteed to find a global minimum for the nuclear norm LASSO subproblems. Consequently, both formulations will have the same number of Pareto curve updates, since the derivatives are necessarily equal at any global minimum whenever ρ is strictly convex². Therefore, when the chosen rank of the factors L and R is appropriately large, our factorized formulation will not require additional Pareto curve updates compared to the classic nuclear norm (BPDN $_{\eta}$) formulation.

Note that it may also be possible to take advantage of randomization techniques as proposed in [15], and/or Lanczos methods, to speed up standard SVD-based approaches. However, we do not know of an existing algorithm for rank minimization that implements this strategy. For this reason, in section 8, we compare only with TFOCS [3] for the least squares (BPDN $_{\eta}$) formulation (see Table 8.3).

6. Robust Formulations. Robust statistics [18, 26] play a crucial role in many real-world applications, allowing good solutions to be obtained in spite of data contamination. In the linear and nonlinear regression setting, the least-squares problem

$$\min_X \|F(X) - b\|_2^2$$

corresponds to the maximum likelihood estimate of X for the statistical model

$$b = F(X) + \epsilon, \tag{6.1}$$

where ϵ is a vector of i.i.d. Gaussian variables. Robust statistical approaches relax the Gaussian assumption, allowing other (heavier tailed) distributions to be used. Maximum likelihood estimates of X under these assumptions are more robust to data contamination. Heavy-tailed distributions, in particular the Student's t , yield formulations that are extremely robust to outliers [19, 2]. This can be explained by their re-descending influence functions [26]. The relationship between densities, penalties, and influence functions is shown in figure 6. Assuming that ϵ has the Student's t density leads to the maximum likelihood estimation problem

$$\min_X \rho(F(x) - b) := \sum_i \log(\nu + (F(X)_i - b_i)^2) \tag{6.2}$$

²It is shown in [1] that for any differentiable convex ρ , the dual problem for the residual $r = b - \mathcal{A}x$ has a unique solution. Therefore, *any* global minimum for (LASSO $_{\rho}$) guarantees a unique residual when ρ is strictly convex, and the claim follows, since the derivative only depends on the residual.

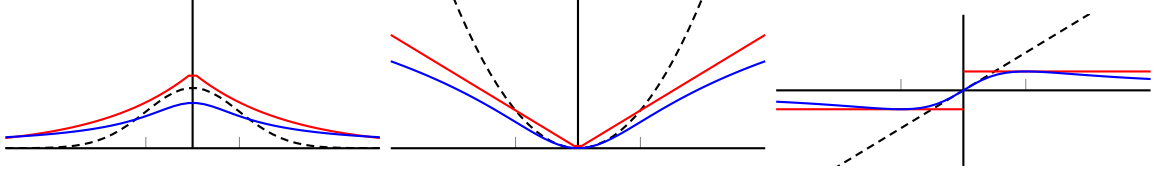


FIG. 6.1. *Gaussian, Laplace, and Student's t Densities, Corresponding Negative Log Likelihoods, and Influence Functions.*

where ν is the Student's t degree of freedom.

A general version of (BPDN $_{\eta}$) was proposed in [1], allowing different penalty functionals ρ . The root-finding procedure of [4] was extended in [1] to this more general context, and used for root finding for both convex and nonconvex ρ (e.g. as in (6.2)).

The use of alternative penalties as constraints in (BPDN $_{\eta}$) does not follow directly from considering statistical models of the form (6.1). However, we can think about penalties ρ as *agents* who, given an error budget η , distribute it between elements of the residual. The strategy that each agent ρ will use to accomplish this task can be deduced from tail features evident in Figure 6. Specifically, the cost of a large residual is prohibitively expensive for the least squares penalty, since its *cost* is commensurate with that of a very large number of small residuals. For example, $(10\alpha)^2 = 100\alpha^2$; so a residual of size 10α is worth as much as 100 residuals of size α to the least squares penalty. Therefore, a least squares penalty will never assign a single residual a relatively large value, since this would quickly use up the entire error budget. In contrast, $|10\alpha| = 10|\alpha|$, so a residual of size 10α is worth only 10 residuals of size α when the 1-norm penalty is used. This penalty is likely to grant a few relatively large errors to certain residuals, if this resulted in a better fit. For the penalty in (6.2), it is easy to see that the cost of a residual of size 10α can be worth fewer than 10 residuals of size α , and specific computations depend on ν and actual size of α . A nonconvex penalty ρ , e.g. the one in (6.2), would happily let outliers have large residuals, as long as the majority of the remaining residuals could stay very small.

From the discussion in the previous paragraph, it is clear that robust penalties are useful as *constraints* in (BPDN $_{\eta}$), and can cleverly distribute the allotted error budget η , using it for outliers while fitting good data. The framework proposed in this paper captures the robust extension, allowing robust data interpolation in situations when some of available data is heavily contaminated. To develop this extension, we follow [1] to define the generalized value function

$$v_{\rho}(\tau) = \min_X \rho(\mathcal{A}(X) - b) \quad \text{s.t.} \quad \|X\|_* \leq \tau, \quad (6.3)$$

and find $v_{\rho}(\tau) = \eta$ using the iteration (2.2). As discussed in section 2, for any *convex* smooth penalty ρ ,

$$v'_{\rho}(\tau) = -\|\mathcal{A}^* \nabla \rho(\bar{r})\|_2, \quad (6.4)$$

where $\|\cdot\|_2$ is the spectral norm, and $\bar{r} = \mathcal{A}(\bar{X}) - b$ for optimal solution \bar{X} that achieves $v_{\rho}(\tau)$. For smooth non-convex ρ , e.g. (6.2), we still use (6.4) in iteration (2.2).

Just as in the standard least squares case, we use the factorization formulation to avoid SVDs. Note that Theorem 4.1 and Corollary 4.2 hold for any choice of penalty ρ . When evaluating the value function $v_{\rho}(\tau)$, we actually solve

$$\min_{L,R} \rho(\mathcal{A}(LR^T) - b) \quad \text{s.t.} \quad \frac{1}{2} \left\| \begin{bmatrix} L \\ R \end{bmatrix} \right\|_F^2 \leq \tau. \quad (6.5)$$

For any smooth penalty ρ , including (6.2), this problem can be solved using the projected gradient method.

7. Reweighting. Every rank- k solution \bar{X} of (BPDN $_{\eta}$) lives in a lower dimensional subspace of $\mathbb{R}^{n \times m}$ spanned by the $n \times k$ row and $m \times k$ column basis vectors corresponding to the nonzero singular values of \bar{X} . In certain situations, it is possible to estimate the row and column subspaces of the matrix X either from prior subspace information or by solving an initial (BPDN $_{\eta}$) problem.

In the case where X is a vector, it was shown that prior information on the support (nonzero entries) of X can be incorporated in the ℓ_1 -recovery algorithm by solving the weighted- ℓ_1 minimization problem. In this case, the weights are applied such that solutions with large nonzero entries on the support estimate have a lower cost (weighted ℓ_1 norm) than solutions with large nonzeros outside of the support estimate [12].

When X is a matrix, the support estimate is replaced by estimates of the row and column subspace bases $U_0 \in \mathbb{R}^{n \times k}$ and $V_0 \in \mathbb{R}^{m \times k}$ of the largest k singular values of X . Let the matrices $\tilde{U} \in \mathbb{R}^{n \times k}$ and $\tilde{V} \in \mathbb{R}^{m \times k}$ be estimates of U_0 and V_0 , respectively. The weighted nuclear norm minimization problem can then be formulated as follows:

$$\min_X \|QXW\|_* \quad \text{s.t.} \quad \rho(\mathcal{A}(X) - b) \leq \eta, \quad (\text{wBPDN}_\eta)$$

where $Q = \omega \tilde{U} \tilde{U}' + \tilde{U}^\perp \tilde{U}^{\perp'}$, $W = \omega \tilde{V} \tilde{V}' + \tilde{V}^\perp \tilde{V}^{\perp'}$, and ω is some constant between zero and one. Here, we use the notation $\tilde{U}^\perp \in \mathbb{R}^{n \times n-k}$ to refer to the orthogonal complement of \tilde{U} in $\mathbb{R}^{n \times n}$, and similarly for \tilde{V}^\perp in $\mathbb{R}^{m \times m}$.

Note that matrices Q and W are invertible, and hence the reweighted LASSO problem still fits into the class of problems characterized by Theorem 4.1. Specifically, we can write any objective $f(X)$ subject to a reweighted nuclear norm constraint as

$$\begin{aligned} \min \quad & f(Q^{-1} \mathcal{R}(Z) W^{-1}) \\ \text{s.t.} \quad & \text{Tr}(Z) \leq \tau, \end{aligned} \quad (7.1)$$

where as in Corollary 4.2, $\mathcal{R}(Z)$ extracts the upper $n \times m$ block of Z . A factorization similar to (5.2) can then be formulated for the (wBPDN $_\eta$) problem in order to optimize over the lower dimensional factors $L \in \mathbb{R}^{n \times k}$ and $R \in \mathbb{R}^{m \times k}$.

In particular, we can solve a sequence of (LASSO $_\tau$) problems

$$\min_{L, R} \|\mathcal{A}(LR^T) - b\|_2^2 \quad \text{s.t.} \quad \frac{1}{2} \left\| \begin{bmatrix} QL \\ WR \end{bmatrix} \right\|_F^2 \leq \tau, \quad (7.2)$$

where Q and W are as defined above. Problem (7.2) can also be solved using the spectral projected gradient algorithm. However, unlike to the non-weighted formulation, the projection in this case is nontrivial. Fortunately, the structure of the problem allows us to find an efficient formulation for the projection operator.

7.1. Projection onto the weighted Frobenius norm ball. The projection of a point (L, R) onto the weighted Frobenius norm ball $\frac{1}{2} (\|QL\|_F^2 + \|WR\|_F^2) \leq \tau$ is achieved by finding the point (\tilde{L}, \tilde{R}) that solves

$$\min_{\tilde{L}, \tilde{R}} \frac{1}{2} \left\| \begin{bmatrix} \hat{L} - L \\ \hat{R} - R \end{bmatrix} \right\|_F^2 \quad \text{s.t.} \quad \frac{1}{2} \left\| \begin{bmatrix} Q\hat{L} \\ W\hat{R} \end{bmatrix} \right\|_F^2 \leq \tau.$$

The solution to the above problem is given by

$$\begin{aligned} \tilde{L} &= \left((\mu\omega^2 + 1)^{-1} \tilde{U} \tilde{U}^T + (\mu + 1)^{-1} \tilde{U}^\perp \tilde{U}^{\perp T} \right) L \\ \tilde{R} &= \left((\mu\omega^2 + 1)^{-1} \tilde{V} \tilde{V}^T + (\mu + 1)^{-1} \tilde{V}^\perp \tilde{V}^{\perp T} \right) R, \end{aligned}$$

where μ is the Lagrange multiplier that solves $f(\mu) \leq \tau$ with $f(\mu)$ given by

$$\begin{aligned} \frac{1}{2} \text{Tr} \left[\left(\frac{\omega^2}{(\mu\omega^2 + 1)^2} \tilde{U} \tilde{U}^T + \frac{1}{(\mu + 1)^2} \tilde{U}^\perp \tilde{U}^{\perp T} \right) LL^T \right. \\ \left. + \left(\frac{\omega^2}{(\mu\omega^2 + 1)^2} \tilde{V} \tilde{V}^T + \frac{1}{(\mu + 1)^2} \tilde{V}^\perp \tilde{V}^{\perp T} \right) RR^T \right]. \end{aligned}$$

The optimal μ that solves equation (7.1) can be found using the Newton iteration

$$\mu^{(t)} = \mu^{(t-1)} - \frac{f(\mu^{(t-1)}) - \tau}{\nabla f(\mu^{(t-1)})},$$

where $\nabla f(\mu)$ is given by

$$\begin{aligned} & \text{Tr} \left[\left(\frac{-2\omega^4}{(\mu\omega^2 + 1)^2} \tilde{U} \tilde{U}^T + \frac{-2}{(\mu + 1)^3} \tilde{U}^\perp \tilde{U}^{\perp T} \right) L L^T \right. \\ & \left. + \left(\frac{-2\omega^4}{(\mu\omega^2 + 1)^3} \tilde{V} \tilde{V}^T + \frac{-2}{(\mu + 1)^3} \tilde{V}^\perp \tilde{V}^{\perp T} \right) R R^T \right]. \end{aligned}$$

7.2. Traversing the Pareto curve. The design of an effective optimization method that solves (wBPDN $_{\eta}$) requires 1) evaluating problem (7.2), and 2) computing the dual of the weighted nuclear norm $\|QXW\|_*$.

To compute the dual of the weighted nuclear norm, we follow Theorem 5.1 of [5] which defines the polar (or dual) representation of a weighted gauge function $\kappa(\Phi x)$ as $\kappa^o(\Phi^{-1}x)$, where Φ is an invertible linear operator. The weighted nuclear norm $\|QXW\|_*$ is in fact a gauge function with invertible linear weighting matrices Q and W . Therefore, the dual norm is given by

$$(\|Q(\cdot)W\|_*)_d(Z) := \|Q^{-1}ZW^{-1}\|_{\infty}.$$

8. Numerical experiments. We test the performance of our algorithm on two example applications. In section 8.1, we consider the Netflix Prize problem, which is often solved using rank minimization [13, 14, 30]. Using this problem, we compare and discuss advantages of regularization *formulations*, as opposed to algorithms; in particular we emphasize the advantages the (BPDN $_{\eta}$) formulation from a modeling perspective.

In section 8.2, we apply the proposed methods and extensions to seismic trace interpolation, a key application in exploration geophysics [33], where rank regularization approaches have recently been successfully used [28]. We compare with classic SVD-based approaches for (BPDN $_{\eta}$) formulation in section 8.2.1, show results for robust completion in section 8.2.2, and present results for the weighted extension in section 8.2.3.

8.1. The Netflix problem. We tested the performance of our algorithm on the MovieLens (1M) dataset, which contains 1,000,209 anonymous ratings of approximately 3,900 movies made by 6,040 MovieLens users. The ratings are on an integer scale from 1 to 5. The ratings matrix is not complete, and the goal is to infer the values in the unseen test set. In order to test our algorithm, we further subsampled the available ratings by randomly removing 50% of the known entries. We then solved the (BPDN $_{\eta}$) formulation to complete the matrix, and compared the predicted (P) and actual (A) removed entries in order to assess algorithm performance. We report the signal-to-noise ratio (SNR):

$$\text{SNR} = 20 \log \left(\frac{\|A\|}{\|P - A\|} \right)$$

for different values of η in the (BPDN $_{\eta}$) formulation.

Since our algorithm requires pre-defining the rank of the factors L and R ³, we perform the recovery with the rank $k \in \{5, 10, 30, 50\}$. Table 8.1 shows the reconstruction SNR for each of the ranks k and for a relative error $\eta \in \{0.5, 0.3, 0.2\}$ (the data mismatch is reduced to a fraction η of the initial error). The last row of table 8.1 shows the recovery for the unconstrained factorized formulation obtained using LSQR; this corresponds to $\eta \approx 0$. It is clear that for small k , we get good results without regularization; however, as k increases, the quality of the LSQR results quickly decay. The best results are obtained by picking a reasonably high k and using regularization. This observation demonstrates the importance of the nuclear norm regularization, especially when the underlying rank of the problem is unknown.

Table 8.2 shows the value of $\|X\|_*$ of the reconstructed signal corresponding to the settings in Table 8.1. While the interpretation of the η values are straightforward (they are fractions of the initial data error), it is much more difficult to predict ahead of time which value of τ one may want to use when solving (LASSO $_{\tau}$). This illustrates the *modeling* advantage of the (BPDN $_{\eta}$) formulation: it requires only the simple parameter η . Once η is provided, the algorithm (not the user) will instantiate (LASSO $_{\tau}$) formulations, and find the right value τ that satisfies $v(\tau) = \eta$.

³ The algorithm also requires the variables L, R to be initialized to non-zero values (unlike the classic SPGL $_1$ algorithm, where $X = 0$ is the default initialization). We initialize them with small Gaussian random values, small enough that the initial τ estimate (taken to be the sum of squares of all entries of L and R) is much smaller than the energy of the data.

TABLE 8.1

Summary of the recovery results (SNR in dB) on the Movielens (1M) data set for factor rank k and relative error level η for (BPDN $_{\eta}$). The last row gives recovery results for the non-regularized data fitting factorized formulation solved with LSQR. Quality degrades with k due to overfitting for the non-regularized formulation, and improves with k when regularization is used.

	k	5	10	30	50
η					
0.5		6.2	7	7.1	6.3
0.3		10.9	10.2	10.7	10.3
0.2		12.2	12.4	12.5	12.6
LSQR ($\eta \approx 0$)		12.2	11.3	8.7	7.8

TABLE 8.2

Nuclear-norms of the solutions $X = LR^T$ for results in Table 8.1, corresponding to τ values in (LASSO $_{\tau}$). These values are found automatically via root finding, but are difficult to guess ahead of time.

	k	5	10	30	50
η					
0.5		6e3	7e3	7e3	6e3
0.3		1.1e4	1.0e4	1.1e4	1.0e4
0.2		2.5e5	0.8e5	0.8e5	0.8e5

8.2. Seismic missing-trace interpolation. In exploration seismology, large-scale data sets (often on the order of petabytes) must be acquired and processed in order to determine the structure of the subsurface. In many situations, only a subset of the complete data is acquired due to physical and/or budgetary constraints. Recent insights from the field of compressed sensing allow for deliberate subsampling of seismic wavefields in order to improve reconstruction quality and reduce acquisition costs [17]. The acquired subset of the complete data is often chosen by randomly subsampling a dense regular source or receiver grid. Interpolation algorithms are then used to reconstruct the dense regular grid in order to perform additional processing on the data such as removal of artifacts, improvement of spatial resolution, and key analysis, such as migration.

In this section, we apply the new rank-minimization approach, along with weighted and robust extensions, to the trace interpolation problem for two different seismic acquisition examples. We first describe the structure of the datasets, and then present the transform we use to cast the interpolation as a rank-minimization problem.

The first example is a real data example from the Gulf of Suez. Seismic data are organized into *seismic lines*, where N_r receivers and N_s sources are collocated in a straight line. Sources are deployed sequentially, and receivers record each shot record for a period of N_t time samples. The Gulf of Suez data contains $N_s = 354$ sources, $N_r = 354$ receivers, and $N_t = 1024$ with a sampling interval of 0.004s, leading to a shot duration of 4s and a maximum temporal frequency of 125 Hz. Most of the energy of the seismic line is preserved when we restrict the spectrum to the 12-60Hz frequency band. Figs. 8.1(a) and (b) illustrate the 12Hz and 60Hz frequency slices in the source-receiver domain, respectively. In order to simulate missing traces, we apply a subsampling mask that randomly removes 50% of the sources, resulting in the subsampled frequency slices illustrated in Figs. 8.1 (c) and (d).

State of the art trace interpolation schemes transform the data into sparsifying domains, for example using the Fourier [33] and Curvelet [17] transforms. The underlying *sparse structure* of the data is then exploited to recover the missing traces. The approach proposed in this paper allows us to instead exploit the *matrix structure* of seismic data, and to design formulations that can achieve trace interpolation using matrix-completion strategies.

The main challenge in applying rank-minimization for seismic trace interpolation is to find a transform domain that satisfies the following two properties:

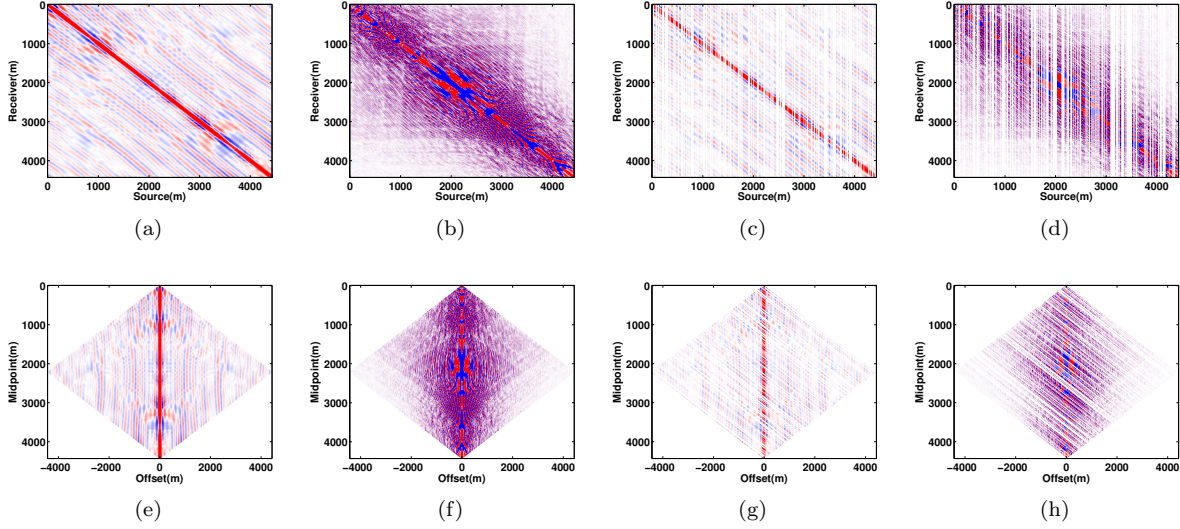


FIG. 8.1. Frequency slices of a seismic line from Gulf of Suez with 354 shots, 354 receivers. Fully data for (a) low frequency at 12 Hz and (b) high frequency at 60 Hz in s-r domain. 50% Subsampled data for (c) low frequency at 12 Hz and (d) high frequency at 60 Hz in s-r domain. Full data for (e) low frequency at 12 Hz and (f) high frequency at 60 Hz in m-h domain. 50% subsampled data for (g) low frequency at 12 Hz and (h) high frequency at 60 Hz in m-h domain.

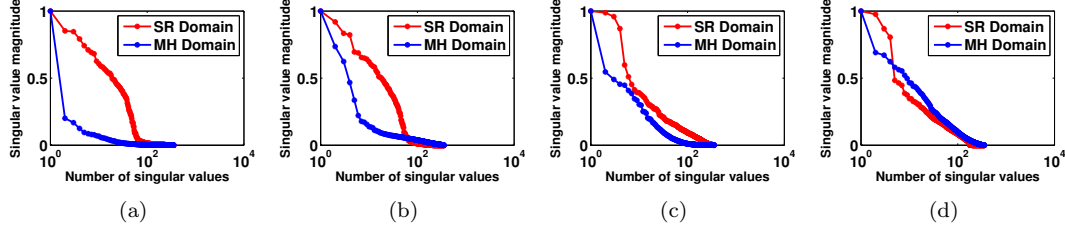


FIG. 8.2. Singular value decay of fully sampled (a) low frequency slice at 12 Hz and (c) high frequency slice at 60 Hz in (s-r) and (m-h) domains. Singular value decay of 50% subsampled (b) low frequency slice at 12 Hz and (d) high frequency data at 60 Hz in (s-r) and (m-h) domains. Notice that for both high and low frequencies, decay of singular values is faster in the fully sampled (m-h) domain than in the fully sampled (s-r) domain, and that subsampling does not significantly change the decay of singular value in (s-r) domain, while it destroys fast decay of singular values in (m-h) domain.

1. Fully sampled seismic lines have low-rank structure (quickly decaying singular values)
2. Subsampled seismic lines have high rank (slowly decaying singular values).

When these two properties hold, rank-penalization formulations allow the recovery of missing traces. To achieve these aims, we use the transformation from the source-receiver (s-r) domain to the midpoint-offset (m-h). The conversion from (s-r) domain to (m-h) domain is a coordinate transformation, with the midpoint is defined by $m = \frac{1}{2}(s + r)$ and the half-offset is defined by $h = \frac{1}{2}(s - r)$. This transformation is illustrated by transforming the 12Hz and 60Hz source-receiver domain frequency slices in Figs. 8.1(a) and (b) to the midpoint-offset domain frequency slices in Figs. 8.1(e) and (f). The corresponding subsampled frequency slices in the midpoint-offset domain are shown in Figs. 8.1(g) and (h).

To show that the midpoint-offset transformation achieves aims 1 and 2 above, we plot the decay of the singular values of both the 12Hz and 60Hz frequency slices in the source-receiver domain and in the midpoint-offset domain in Figs. 8.2 (a) and (c). Notice that the singular values of both frequency slices decay faster in the midpoint-offset domain, and that the singular value decay is slower for subsampled data in Figs. 8.2 (b) and (d).

Let \mathbf{X} denote the data matrix in the midpoint-offset domain and let \mathbf{R} be the subsampling operator that maps Figs. 8.1 (e) and (f) to Figs. 8.1(g) and (h). Denote by \mathcal{S} the transformation operator from the source-

receiver domain to the midpoint-offset domain. The resulting measurement operator in the midpoint-offset domain is then given by $\mathcal{A} = \mathbf{R}\mathbf{S}^{\mathbf{H}}$.

We formulate and solve the matrix completion problem (BPDN_{η}) to recover a seismic line from Gulf of Suez in the (m-h) domain. We work with monochromatic frequency slices and adjust the rank while going from low to high frequency slices. We use 300 iterations of SPGL_1 for all frequency slices. Figures 8.3(a) and (b) show the recovery and error plot for the low frequency slice at 12 Hz, respectively. Figures 8.3(c) and (d) show the recovery and error plot for the high frequency slice at 60 Hz, respectively. Figure 8.4 shows a common-shot gather section after missing-trace interpolation from Gulf of Suez data set.

In the second acquisition example, we implement the proposed formulation on the 5D synthetic seismic data provided by BG. We extract a single 4D monochromatic frequency slice from this data set and perform the missing-trace interpolation. Each monochromatic frequency slice has 400×400 receivers spaced by 25m and 68×68 sources spaced by 150m. To obtain the observed data, we apply sub-sampling masks that randomly remove 25% and 50% of the shots. In case of 4D, we have two choices of matricization, as shown in Figures 8.5(a,b), where we can either place the (ReceiverX,ReceiverY) dimensions in the rows and (SourceX,SourceY) dimensions in the columns, or (ReceiverY,SourceY) dimensions in the rows and (ReceiverX,SourceX) dimensions in the columns. We observed the same behavior of singular value decay as shown in Figure 8.5(e,f), for each of these strategies. We therefore selected the transform domain to be the permutation of source and receivers coordinates, where matricization of each 4D monochromatic frequency slices is done using (SourceX, ReceiverX) and (SourceY, ReceiverY) coordinates. We use rank 120 for the interpolation, and run the solver for a maximum of 1000 iterations. Figures 8.6 and 8.7 show the interpolation results for 25% and 50% subsampling ratios, respectively. The resulting interpolations have low reconstruction error.

In order to illustrate the importance of the nuclear-norm regularization, we solved the interpolation problem using a simple least-squares formulation on the same seismic data set from Gulf of Suez. The least squares problem was solved using the L , R factorization structure, thereby implicitly enforcing a rank on the recovered estimate (i.e, formulation (5.2) was optimized without the τ -constraint). The problem was then solved with the factors L and R having a rank $k \in \{5, 10, 20, 30, 40, 50, 80, 100\}$. The reconstruction SNRs comparing the recovery for the regularized and non-regularized formulations are shown in Fig. 8.8. The figure shows that the performance of the non-regularized approach decays with rank, due to overfitting. The regularized approach, in contrast, obtains better recovery as the factor rank increases.

8.2.1. Comparison with classical nuclear-norm formulation. To illustrate the advantage of proposed matrix-factorization formulation (which we refer to as LR below) over classical nuclear-norm formulation, we compare the reconstruction error and computation time with the existing techniques. The most natural baseline is the SPGL_1 algorithm [5] applied to the classic nuclear norm (BPDN_{η}) formulation, where the decision variable is X , the penalty function ρ is the 2-norm, and the projection is done using the SVD. This example tests the classic formulation against LR, in the same algorithmic framework. The second comparison is with the TFOCS[3] algorithm, which can also solve the (BPDN_{η}) formulation.

The comparisons are done using two different data sets. In the first example, we generated a rank 20 matrix of size 100×100 . We subsampled the matrix by randomly removing 50% of the data entries. The synthetic low-rank example in table 8.3 shows the comparison of SNR and computational time. The rank of the factors was set to 20, the true rank of the data matrix, in this experiment. Both the classic SPGL_1 algorithm and the LR reformulation are faster than TFOCS. The speed advantage increases as the error threshold η decreases. LR is faster than the classic SPGL_1 , and this advantage also increases as the data is required to be fit more tightly. The standard SPGL_1 formulation and TFOCS produce roughly the same quality of result for each iteration. LR gives uniformly better SNR results, and the improvement is very significant for lower error thresholds. The high quality of these results are due to the fact that the rank of the factors was set to the true rank of the data matrix⁴.

In the second example, we interpolated missing traces of a monochromatic frequency slice (of size 354×354), extracted from Gulf of Suez data set. We subsampled the frequency slice by randomly removing the

⁴We tested this hypothesis by re-running the experiment with higher factor rank. For example, selecting factor rank to be 40 gives SNRs of 15.7, 28.5, 36.7, 44.1 for the corresponding η values for the synthetic low-rank experiment in 8.3. These values are lower than when the true rank is used, but still *uniformly better* than those returned by SPGL_1 , as we would expect.

50% of shots and performed the missing-trace interpolation in (m-h) domain. We compare the SNR and computation time for a fixed set of iterations and η . The rank of the factors was set to 35. The seismic example in table 8.3 shows the results. Once again, both the classic SPGL_1 algorithm and LR are faster than TFOCS. Moreover, on this larger example LR is also significantly faster than SPGL_1 . Note that our implementation of LR does not take advantage of the clever residual formation idea discussed in section 5.2, since we form the full matrix LR^T each time to compute the residual. Implementing a routine that only computes the entries required to compare with b would introduce additional savings (in this case, we would expect the LR formulation to be twice as fast). In the quality of recovery, both SPGL_1 and LR have better SNR than TFOCS. LR achieves lower SNR than SPGL_1 for high error thresholds, but better results for lower thresholds.

TABLE 8.3

*TFOCS versus classic SPGL_1 versus LR factorization. **Synthetic low rank** example shows results for completing a rank $20 \times 100 \times 100$ matrix, with 50% missing entries. Computational time and SNR are shown for $\eta = 0.1, 0.01, 0.001, 0.0001$. Rank of the factors is taken to be 20. **Seismic** example shows results for matrix completion a low-frequency slice at 10 Hz, extracted from the Gulf of Suez data set, with 50% missing entries. Computational time and SNR are shown for $\eta = 0.2, 0.1, 0.05, 0.04$. Rank of factors was taken to be 35.*

		Synthetic low rank						Seismic			
η		0.1	0.01	0.001	0.0001	η		0.2	0.1	0.05	0.04
TFOCS	SNR (dB)	15.5	24.7	25.7	25.8	TFOCS	SNR (dB)	9.5	11	11.3	11.4
	<i>time (s)</i>	<i>42</i>	<i>298</i>	<i>686</i>	<i>726</i>		TFOCS	<i>time (s)</i>	<i>39</i>	<i>140</i>	<i>460</i>
$\text{SPG}l_1$	SNR (dB)	15.7	24.7	25.7	24.1	$\text{SPG}l_1$		SNR (dB)	12.8	17.0	19.1
	<i>time (s)</i>	<i>1.4</i>	<i>3.9</i>	<i>9.5</i>	<i>6.9</i>		$\text{SPG}l_1$	<i>time (s)</i>	<i>24.6</i>	<i>34.6</i>	<i>56.2</i>
LR	SNR (dB)	15.8	31.9	49.0	62.8	LR		SNR (dB)	10.1	15.0	19.7
	<i>time (s)</i>	<i>0.6</i>	<i>0.7</i>	<i>5.3</i>	<i>3.9</i>		LR	<i>time (s)</i>	<i>0.85</i>	<i>1.3</i>	<i>8.5</i>

8.2.2. Simultaneous missing-trace interpolation and denoising. To illustrate the utility of robust cost functions, we consider a situation where observed data are heavily contaminated. The goal here is to simultaneously denoise and interpolate the data. We work with same seismic line from Gulf of Suez. To obtain the observed data, we apply a sub-sampling mask that randomly removes 50% of the shots, and to simulate contamination, we replace another 10% of the shots with large random errors, whose amplitudes are three times the maximum amplitude present in the data. In this example, we formulate and solve the robust matrix completion problem (BPDN_η), where the cost ρ is taken to be the penalty (6.2); see section 6 for the explanation and motivation. As in the previous examples, the recovery is done in the (m-h) domain. We implement the formulation in the frequency domain, where we work with monochromatic frequency slices, and adjust the rank and ν parameter while going from low to high frequency slices. Figure 8.9 compares the recovery results with and without using a robust penalty function. The error budget plays a significant role in this example, and we standardized the problems by setting the *relative error* to be 20% of the initial error, so that the formulations are comparable.

We can clearly see that the standard least squares formulation is unable to recover the true solution. The intuitive reason is that the least squares penalty is simply unable to budget large errors to what should be the outlying residuals. The Student's t penalty, in contrast, achieves a good recovery in this extreme situation, with an SNR of 17.9 DB. In this example, we used 300 iterations of SPGL_1 for all frequency slices.

8.2.3. Re-Weighting. Re-weighting for seismic trace interpolation was recently used in [23] to improve the interpolation of subsampled seismic traces in the context of sparsity promotion in the curvelet domain. The weighted ℓ_1 formulation takes advantage of curvelet support overlap across adjacent frequency slices.

Analogously, in the matrix setting, we use the weighted rank-minimization formulation (wBPDN_η) to take advantage of correlated row and column subspaces for adjacent frequency slices. We first demonstrate the effectiveness of solving the (wBPDN_η) problem when we have accurate subspace information. For this purpose, we compute the row and column subspace bases of the fully sampled low frequency (11Hz) seismic slice and pass this information to (wBPDN_η) using matrices Q and W . Figures 8.10(a) and (b) show the

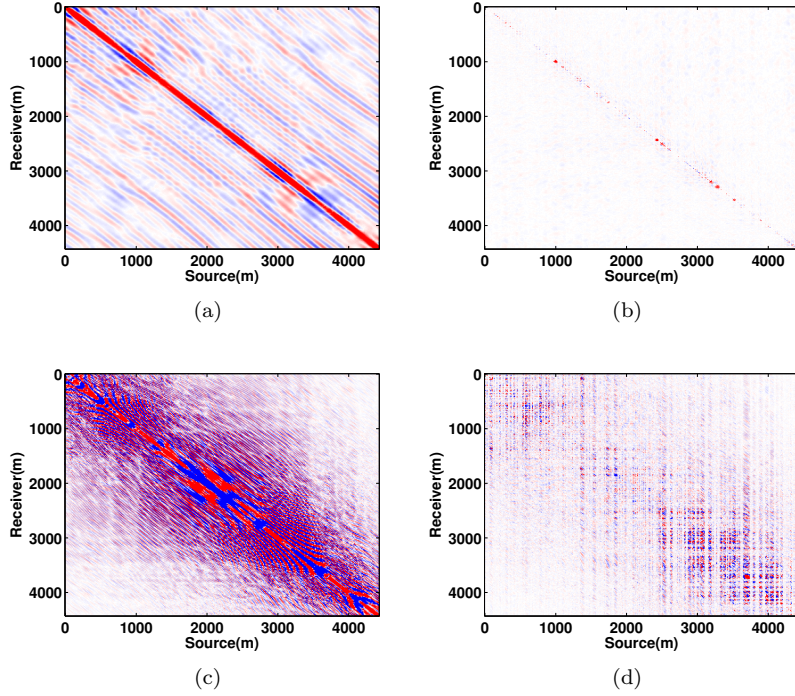


FIG. 8.3. Recovery results for 50% subsampled 2D frequency slices using the nuclear norm formulation. (a) Interpolation and (b) residual of low frequency slice at 12 Hz with SNR = 19.1 dB. (c) Interpolation and (d) residual of high frequency slice at 60 Hz with SNR = 15.2 dB.

residual of the frequency slice with and without weighting. The reconstruction using the (wBPDN_η) problem achieves a 1.5dB improvement in SNR over the non-weighted (BPDN_η) formulation.

Next, we apply the (wBPDN_η) formulation in a practical setting where we do not know subspace bases ahead of time, but learn them as we proceed from low to high frequencies. We use the row and column subspace vectors recovered using (BPDN_η) for 10.75 Hz and 15.75 Hz frequency slices as subspace estimates for the adjacent higher frequency slices at 11 Hz and 16 Hz. Using the (wBPDN_η) formulation in this way yields SNR improvements of 0.6dB and 1dB, respectively, over (BPDN_η) alone. Figures 8.11(a) and (b) show the residual for the next higher frequency without using the support and Figures 8.11(c) and (d) shows the residual for next higher frequency with support from previous frequency. Figure 8.12 shows the recovery SNR versus frequency for weighted and non-weighted cases for a range of frequencies from 9 Hz to 17 Hz.

9. Conclusions. We have presented a new method for matrix completion. Our method combines the Pareto curve approach for optimizing (BPDN_η) formulations with SVD-free matrix factorization methods.

We demonstrated the modeling advantages of the (BPDN_η) formulation on the Netflix Prize problem, and obtained high-quality reconstruction results for the seismic trace interpolation problem. Comparison with state of the art methods for the (BPDN_η) formulation showed that the factorized formulation is faster than both TFOCS and classic SPGL₁ formulations that rely on the SVD.

We also proposed two extensions. First, using robust penalties ρ in (BPDN_η), we showed that simultaneous interpolation and denoising can be achieved in the extreme data contamination case, where 10% of the data was replaced by large outliers. Second, we proposed a weighted extension (wBPDN_η), and used it to incorporate subspace information we learned on the fly to improve interpolation in adjacent frequencies.

10. Appendix.

Proof of Theorem 4.1. Recall [6, Lemma 2.1]: if $SS^T = KK^T$, then $S = KQ$ for some orthogonal matrix $Q \in \mathbb{R}^{r \times r}$. Next, note that the objective and constraints of (4.2) are given in terms of SS^T , and for any orthogonal $Q \in \mathbb{R}^{r \times r}$, we have $SQQ^TS^T = SS^T$, so \tilde{S} is a local minimum of (4.2) if and only if $\tilde{S}Q$ is a

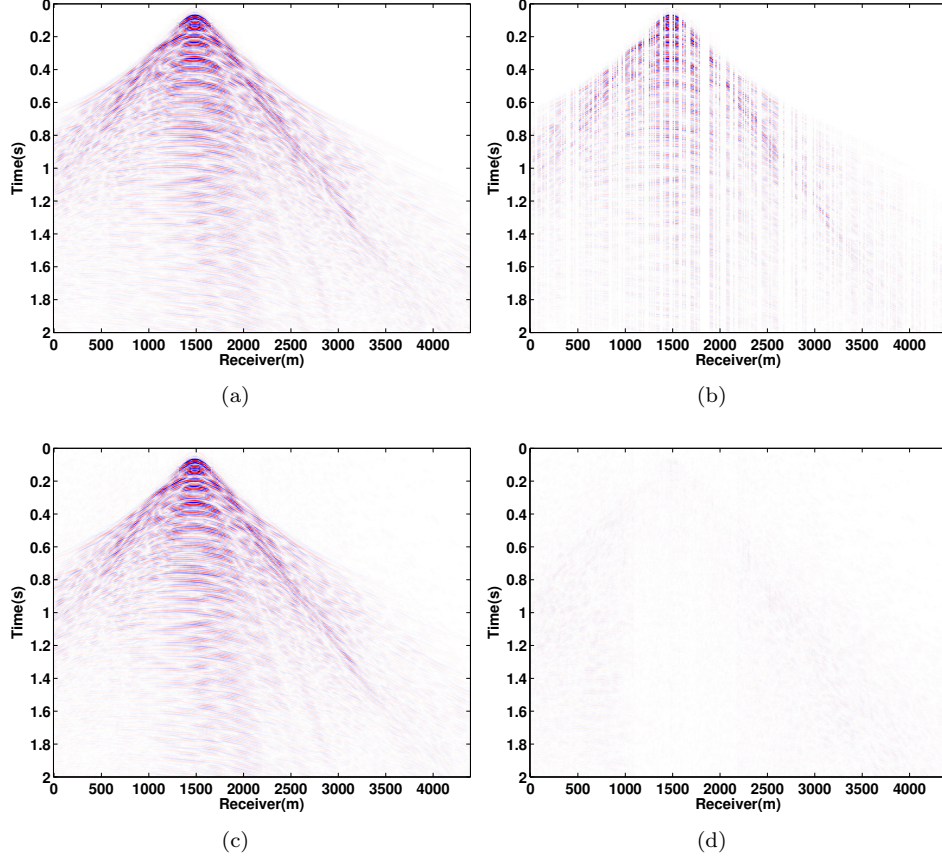


FIG. 8.4. Missing trace interpolation of a seismic line from Gulf of Suez. (a) Ground truth. (b) 50% subsampled common shot gather. (c) Recovery result with a SNR of 18.5 dB. (d) Residual.

local minimum for all orthogonal $Q \in \mathbb{R}^{r \times r}$.

If \bar{Z} is a local minimum of (4.1), then any factor \bar{S} with $\bar{Z} = \bar{S}\bar{S}^T$ is a local minimum of (4.2). Otherwise, we can find a better solution \tilde{S} in the neighborhood of \bar{S} , and then $\tilde{Z} := \tilde{S}\tilde{S}^T$ will be a feasible solution for (4.1) in the neighborhood of \bar{Z} (by continuity of the map $S \rightarrow SS^T$).

We prove the other direction by contrapositive. If \bar{Z} is not a local minimum for (4.1), then you can find a sequence of feasible solutions Z_k with $f(Z_k) < f(\bar{Z})$ and $Z_k \rightarrow \bar{Z}$. For each k , write $Z_k = S_k S_k^T$. Since Z_k are all feasible for (4.1), so S_k are feasible for (4.2). By assumption $\{Z_k\}$ is bounded, and so is S_k ; we can therefore find a subsequence of $S_j \rightarrow \tilde{S}$ with $\tilde{S}\tilde{S}^T = \bar{Z}$, and $f(S_j S_j^T) < f(\tilde{S}\tilde{S}^T)$. In particular, we have $\bar{Z} = \tilde{S}\tilde{S}^T = \tilde{S}\tilde{S}^T$, and \tilde{S} is not a local minimum for (4.2), and therefore (by previous results) \bar{S} cannot be either.

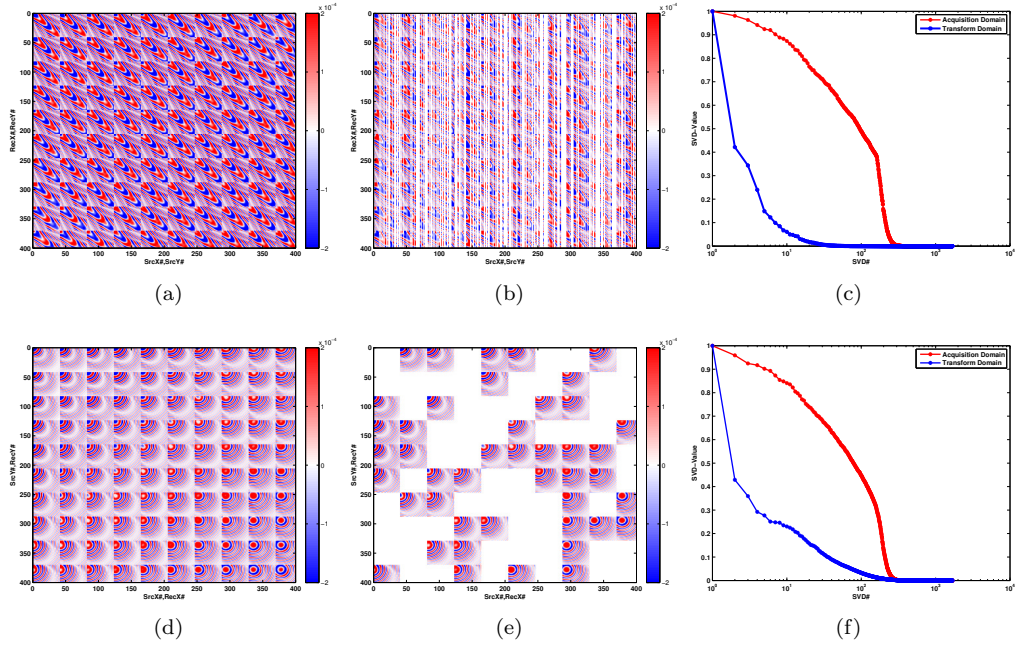


FIG. 8.5. Matricization of 4D monochromatic frequency slice. Top: $(SourceX, SourceY)$ matricization. Bottom: $(SourceX, ReceiverX)$ matricization. Left: Fully sampled data; Middle: Subsampled data; Right: Singular value decay.

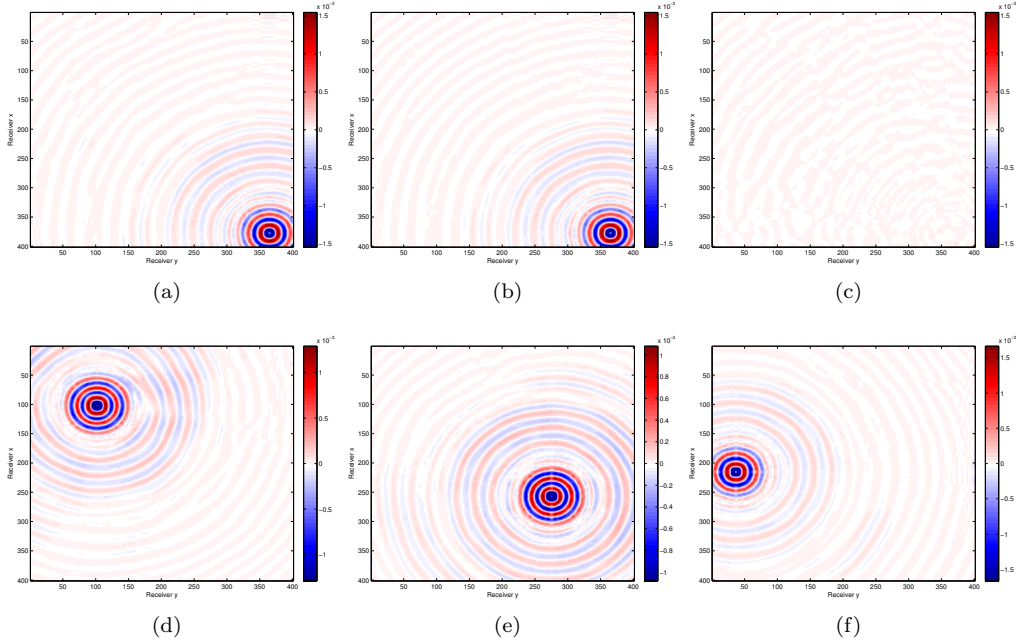


FIG. 8.6. Missing-trace interpolation of a monochromatic frequency slice extracted from 5D data set, subsampling ratio = 25%. (a,b,c) Original, recovery and residual of a common shot gather with a SNR of 24.8 dB at the location where shot is recorded. (e,f,g) Interpolation of common shot gathers at the location where no reference shot is present.

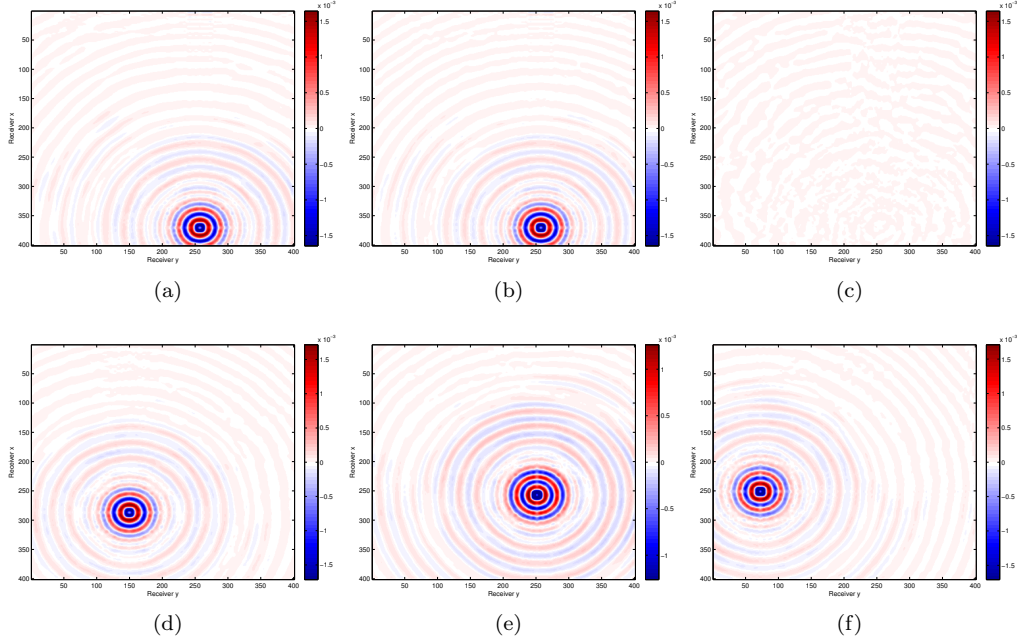


FIG. 8.7. Missing-trace interpolation of a monochromatic frequency slice extracted from 5D data set, subsampling ratio = 50%. (a,b,c) Original, recovery and residual of a common shot gather with a SNR of 25.3 dB at the location where shot is recorded. (e,f,g) Interpolation of common shot gathers at the location where no reference shot is present.

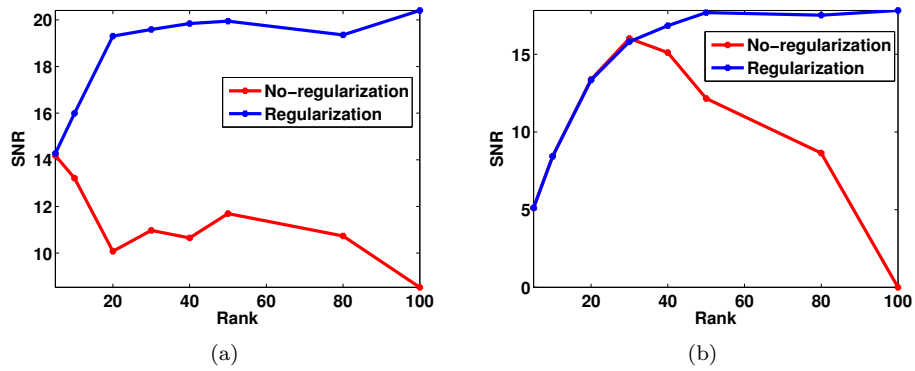


FIG. 8.8. Comparison of regularized and non-regularized formulations. SNR of (a) low frequency slice at 12 Hz and (b) high frequency slice at 60 Hz over a range of factor ranks. Without regularization, recovery quality decays with factor rank due to over-fitting; the regularized formulation improves with higher factor rank.

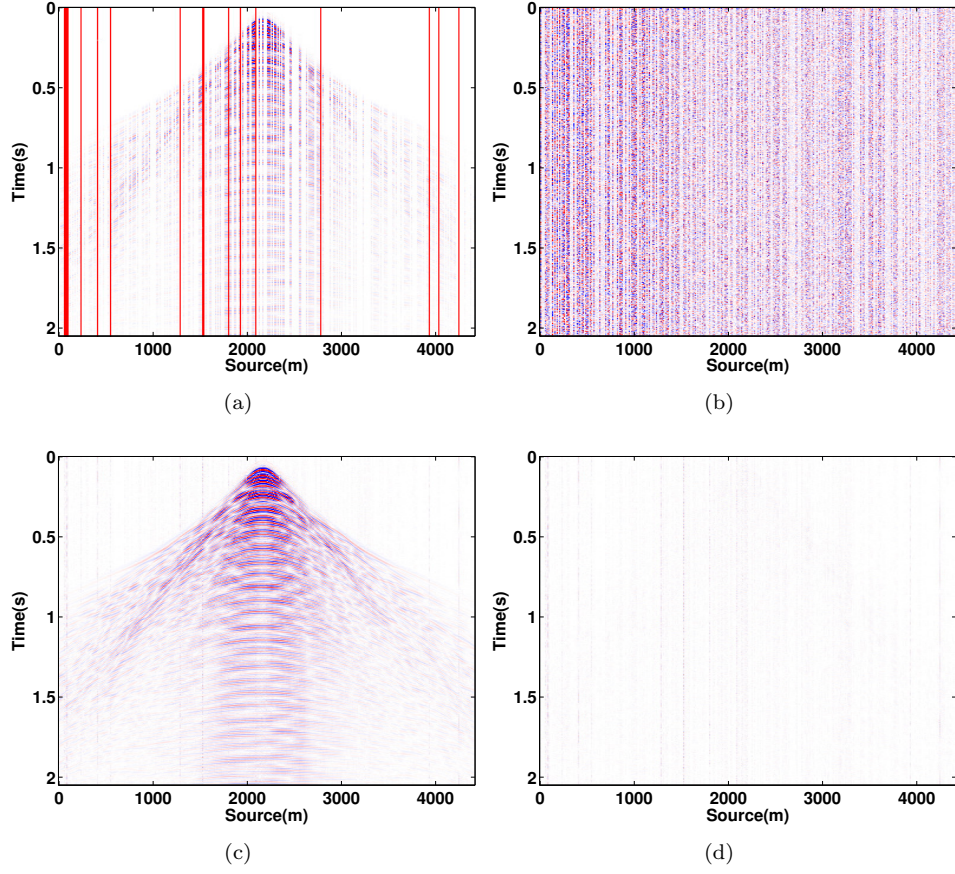


FIG. 8.9. Comparison of interpolation and denoising results for the Student's t and least-squares misfit function. (a) 50% subsampled common receiver gather with another 10 % of the shots replaced by large errors. (b) Recovery result using the least-squares misfit function. (c,d) Recovery and residual results using the student's t misfit function with a SNR of 17.2 dB.

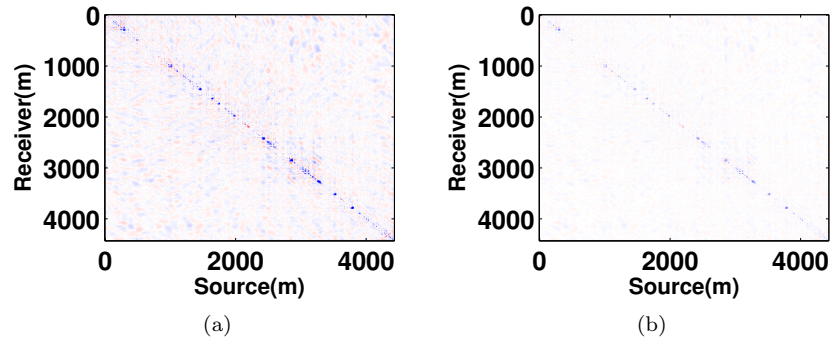


FIG. 8.10. Residual error for recovery of 11 Hz slice (a) without weighting and (b) with weighting using true support. SNR in this case is improved by 1.5 dB.

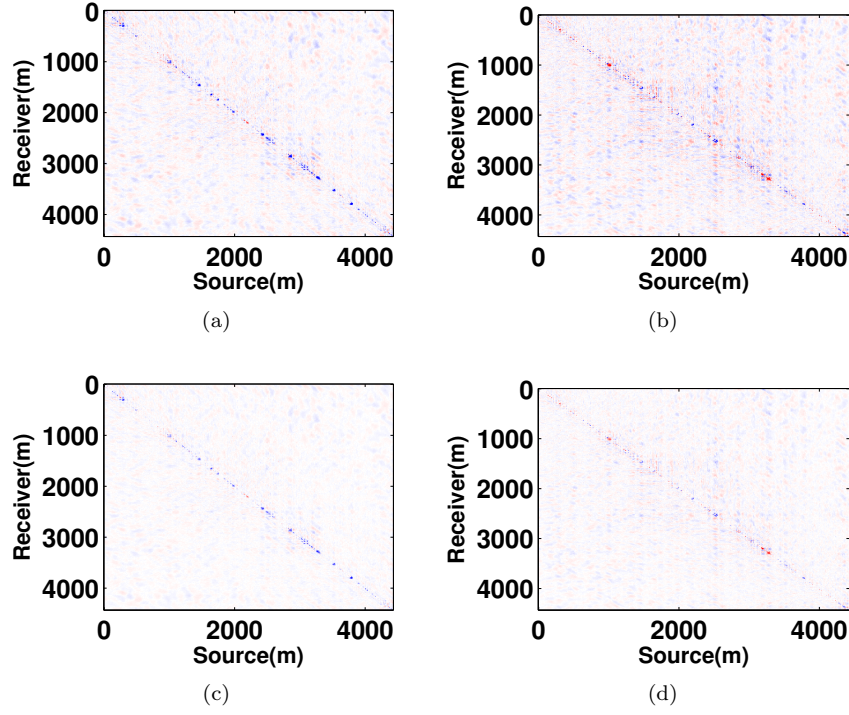


FIG. 8.11. Residual of low frequency slice at 11 Hz (a) without weighing (c) with support from 10.75 Hz frequency slice. SNR is improved by 0.6 dB. Residual of low frequency slice at 16 Hz (b) without weighing (d) with support from 15.75 Hz frequency slice. SNR is improved by 1dB. Weighting using learned support is able to improve on the unweighted interpolation results.

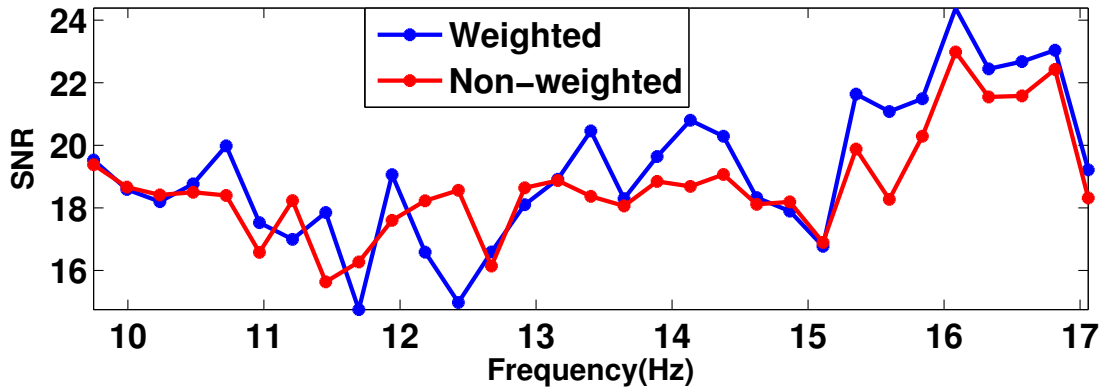


FIG. 8.12. Recovery results of practical scenario in case of weighted factorized formulation over a frequency range of 9-17 Hz. The weighted formulation outperforms the non-weighted for higher frequencies. For some frequency slices, the performance of the non-weighted algorithm is better, because the weighted algorithm can be negatively affected when the subspaces are less correlated.

REFERENCES

- [1] A.Y. ARAVKIN, J.V. BURKE, AND M.P. FRIEDLANDER, *Variational properties of value functions*, Provisionally accepted to SIAM J. Opt., arXiv:1211.3724, (2012).
- [2] A.Y. ARAVKIN, M.P. FRIEDLANDER, F. HERRMANN, AND T. VAN LEEUWEN, *Robust inversion, dimensionality reduction, and randomized sampling*, Mathematical Programming, 134 (2012), pp. 101–125.
- [3] STEPHEN R. BECKER, EMMANUEL J. CANDÈS, AND MICHAEL C. GRANT, *Templates for convex cone problems with applications to sparse signal recovery*, Mathematical Programming Computation, 3 (2011), pp. 165–218.
- [4] E. VAN DEN BERG AND M. P. FRIEDLANDER, *Probing the pareto frontier for basis pursuit solutions*, SIAM Journal on Scientific Computing, 31 (2008), pp. 890–912.
- [5] ———, *Sparse optimization with least-squares constraints*, SIAM J. Optimization, 21 (2011), pp. 1201–1229.
- [6] SAMUEL BURER AND RENATO D. C. MONTEIRO, *Local minima and convergence in low-rank semidefinite programming*, Mathematical Programming, 103 (2003), p. 2005.
- [7] E. CANDÈS, X. LI, Y. MA, AND J. WRIGHT, *Robust principal component analysis?*, Journal of the ACM, 58 (2011).
- [8] E. J. CANDÈS AND T. TAO, *Near-optimal signal recovery from random projections: Universal encoding strategies*, Information Theory, IEEE Transactions on, 52 (2006), pp. 5406–5425.
- [9] D. DONOHO, *Compressed sensing*, IEEE Transactions on Information Theory, 52 (2006), pp. 1289–1306.
- [10] M. FAZEL, *Matrix rank minimization with applications*, PhD thesis, Stanford University, 2002.
- [11] M.A.T. FIGUEIREDO, R.D. NOWAK, AND S.J. WRIGHT, *Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems*, IEEE Journal of Selected Topics in Signal Processing, 1 (2007), pp. 586–597.
- [12] M. FRIEDLANDER, H. MANSOUR, R. SAAB, AND O. YILMAZ, *Recovering compressively sampled signals using partial support information*, IEEE Transactions on Information Theory, 58 (2011).
- [13] S. FUNK, *Netflix update: Try this at home*, December 2006.
- [14] DAVID GROSS, *Recovering Low-Rank Matrices From Few Coefficients in Any Basis*, IEEE Transactions on Information Theory, 57 (2011), pp. 1548–1566.
- [15] N. HALKO, P. G. MARTINSSON, AND J. A. TROPP, *Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions*, SIAM Rev., 53 (2011), pp. 217–288.
- [16] FELIX J. HERRMANN, MICHAEL P. FRIEDLANDER, AND OZGUR YILMAZ, *Fighting the curse of dimensionality: Compressive sensing in exploration seismology*, Signal Processing Magazine, IEEE, 29 (2012), pp. 88–100.
- [17] F. J. HERRMANN AND G. HENNENFENT, *Non-parametric seismic data recovery with curvelet frames*, Geophysical Journal International, 173 (2008), pp. 233–248.
- [18] P.J. HUBER, *Robust Statistics*, Wiley, 1981.
- [19] KENNETH L. LANGE, RODERICK J. A. LITTLE, AND JEREMY M. G. TAYLOR, *Robust statistical modeling using the t distribution*, Journal of the American Statistical Association, 84 (1989), pp. 881–896.
- [20] J. LEE, B. RECHT, R. SALAKHUTDINOV, N. SREBRO, AND J. TROPP, in Advances in Neural Information Processing Systems, 2010.
- [21] ———, *Practical large-scale optimization for max-norm regularization*, in Advances in Neural Information Processing Systems 23, J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, eds., 2010, pp. 1297–1305.
- [22] J. MAIRAL, M. ELAD, AND G. SAPIRO, *Sparse representation for color image restoration*, IEEE Transactions on Image Processing, 17 (2008), pp. 53–69.
- [23] H. MANSOUR, F. J. HERRMANN, AND O. YILMAZ, *Improved wavefield reconstruction from randomized sampling via weighted one-norm minimization*, submitted to Geophysics, (2012).
- [24] H. MANSOUR, R. SAAB, P. NASIOPOULOS, AND R. WARD, *Color image desaturation using sparse reconstruction*, in Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), March 2010, pp. 778–781.
- [25] HASSAN MANSOUR, HANEET WASON, TIM T.Y. LIN, AND FELIX J. HERRMANN, *Randomized marine acquisition with compressive sampling matrices*, Geophysical Prospecting, 60 (2012), pp. 648–662.
- [26] RICARDO A. MARONNA, DOUGLAS MARTIN, AND YOHAI, *Robust Statistics*, Wiley Series in Probability and Statistics, Wiley, 2006.
- [27] R. NEELAMANI, C. E. KROHN, J. R. KREBS, J. K. ROMBERG, MAX DEFFENBAUGH, AND JOHN E. ANDERSON, *Efficient seismic forward modeling using simultaneous random sources and sparsity*, Geophysics, 75 (2010), pp. WB15–WB27.
- [28] V. OROPEZA AND M. SACCHI, *Simultaneous seismic data denoising and reconstruction via multichannel singular spectrum analysis*, Geophysics, 76 (2011), pp. V25–V32.
- [29] B. RECHT, M. FAZEL, AND P.A. PARRILO, *Guaranteed minimum rank solutions to linear matrix equations via nuclear norm minimization*, SIAM Review, 52 (2010), pp. 471–501.
- [30] B. RECHT AND C. RÉ, *Parallel stochastic gradient algorithms for large-scale matrix completion*, in Optimization Online, 2011.
- [31] J. D. M. RENNIE AND N. SREBRO, in ICML '05 Proceedings of the 22nd international conference on Machine learning.
- [32] R.T. ROCKAFELLAR AND R.J-B. WETS, *Variational Analysis*, vol. 317, Springer, 1998.
- [33] M.D. SACCHI, T.J. ULRYCH, AND C.J. WALKER, *Interpolation and extrapolation using a high-resolution discrete fourier transform*, Signal Processing, IEEE Transactions on, 46 (1998), pp. 31–38.
- [34] J.-L. STARCK, M. ELAD, AND D. DONOHO, *Image decomposition via the combination of sparse representation and a variational approach*, IEEE Transaction on Image Processing, 14 (2005).